Remarks

The applicants have carefully considered the Office action mailed March 20, 2008. In view of the following remarks, reconsideration of the application is respectfully requested.

Claim 1 was rejected as anticipated under 35 U.S.C. § 102 by Brumme (US Pub. No. 2005/0172286). Claim 1 recites a method comprising, *inter alia*, receiving a managed application program interface associated with a managed runtime environment, the managed application program interface and the managed runtime environment not supporting the one or more processor instructions; replacing a portion of the managed application program interface with native code supporting the one or more processor instructions to generate an optimized managed application program interface; and compiling the optimized managed application program interface including the native code using a compiler of a managed runtime environment. As explained in detail below, Brumme fails to describe or suggest all of the recitations of claim 1.

As an initial matter, the applicants wish to point out an apparent misunderstanding of Brumme evidenced in the Office action. It appears that the Office action is premised on the idea that Brumme suggests compiling managed code where (i.e., in the circumstance that) the managed code environment disallows calls to managed code. This premise appears to be based on the description in Paras. [0001] and [0004]. In particular, Brumme states "a managed computing environment compiles managed code into native code for certain code that is deemed inappropriate for the particular managed code environment, where the managed code environment disallows calls to managed code that are deemed inappropriate for the particular … managed code environment." (Para. [0001]). However, the cited paragraphs do not support the premise advanced in the Office action. Rather, Brumme states that a computing device compiles managed code into native code that is executed by a common language runtime. (Para. [0001]). The summary of the invention clarifies that Brumme describes allowing a host of a runtime environment to disallow a call to a method from a managed code caller, not to compiling to native code to address an inability of code to access a managed code environment. (Para. [0009]). Brumme does not describe or suggest that compiling managed code to native code may be done to overcome a code environment's prevention of calls to a managed code. Brumme describes that compiling managed code into native code using a just-in-time compiler is merely how the Common Language Runtime execution system executes managed code. (Para. [0001]-[0004]).

Compiling managed code to native code, as described in Brumme, cannot describe replacing a portion of the code to create optimized code and then compiling the optimized code as recited in claim 1. When compiling managed code to native code, the result is native code that is representative of the original managed code. However, replacing a portion of an original code to generate a modified code and then compiling the modified code generates a compiled output that is representative of the modified code, not the original code. Further, even if compiling to native code, as described in Brumme, is the same as replacing a portion of the managed application program interface, a point that the applicants do not concede, the compiling of Brumme cannot describe both the replacing and compiling as recited in claim 1. Accordingly, not only does Brumme fail to describe or suggest that method recited in claim 1, the process described in Brumme does not provide the same result as claim 1.

Brumme, also, does not describe or suggest a managed application program interface and a managed runtime environment <u>not supporting one or more processor instructions</u>. The Office action indicates that Para. [0025] and FIGS. 1-2 describes a "managed API associated with a managed runtime environment where managed runtime does not support processor instruction." (Page 2, lines 23-24). However, no portion of Brumme suggests that a manage runtime environment does not support processor instructions. Rather, Brumme states "[The programming framework] permits multi-language development and seamless integration by supporting multiple languages and encapsulates the underlying operating system and object model services." (Para. [0025]). The applicants are unable to find description in Brumme that supports the assertion in the Office action that Brumme describes a managed runtime environment that does not support a processor instruction. Further, while applicants admit that Brumme describes an application program interface (API) layer of a programming framework, no portion of Brumme cited in the Office action describes or suggests that the API does not support one or more processor instructions. Brumme describes disallowing calls to managed code that are deemed inappropriate for the particular managed code environment, but Brumme does not describe or suggest (nor does the Office action explain how Brumme describes or suggests) that a managed runtime environment does not support one or more processor instructions.

Accordingly, for at least any one of the forgoing reasons, Brumme fails to describe or suggest at least one of the recitations of claim 1. Therefore, claim 1 and all claims depending therefrom are patentable over the cited reference.

Claims 8 and 16 were rejected under the same reasons as claim 1. The applicants respectfully submit that for at least the forgoing reasons, independent claims 8 and 16 and all claims depending therefrom are in condition for allowance.

Claim 23 was rejected under 35 U.S.C. § 103 as unpatentable over Brumme and what is "commonly known." The Office action does not suggest that common knowledge overcomes the deficiencies noted above in Brumme. Therefore, for at least the reasons described in conjunction with claim 1, claim 23 and all claims depending therefrom are in condition for allowance.

Applicants have reviewed the rejections of the remaining dependent claims and wish to point out for the record the following notes regarding the rejection based on Brumme. However, applicants do not admit that characterizations of Brumme not commented-on are valid.

In rejecting claim 3, the Office action contends that Para. [0042] of Brumme describes a proxy stub associated with instructions generated during installation of managed runtime software. However, Para. [0042] and FIG. 6, which is described in part by Para. [0042], are directed to compile time and runtime of managed code. Para. [0042] does not suggest that any of the operations described therein are performed during installation of managed runtime software.

In rejecting claim 5, the Office action contends that Paras. [0042]-[0045] describe proxy/stub marshalling of a virtual machine. However, neither the cited portions of Brumme nor the entire specification of Brumme even once references marshalling language code.

In rejecting claim 6, the Office action contends that Para. [0014] and FIG. 4 describe generating the processor instruction proxy stub in response to identifying the processor associated with the one or more processor instructions. While the cited paragraph and figure show a resource identifier, none of the cited portions of Brumme described generating a processor instruction proxy stub in response to identifying the processor associated with one or more processor instructions.

Conclusion

The applicants respectfully submit that all claims are in condition for allowance. Reconsideration of the application and allowance thereof are respectfully requested. If there is any matter that the examiner would like to discuss, the examiner is invited to contact the undersigned representative at the telephone number set forth below.

Respectfully submitted,

HANLEY, FLIGHT & ZIMMERMAN, LLC
150 S. Wacker Dr.
Suite 2100
Chicago, Illinois 60606

Dated: **May 20, 2008**

/Michael W. Zimmerman/

Michael W. Zimmerman
Reg. No. 57,993
Agent for Applicants
312.580.1020